

## Caso de estudio: Modelo ARIMA

Los modelos **ARIMA** (por las siglas en inglés de AutoRegressive Moving Average) son una clase popular y flexible de modelos de pronóstico que utilizan información histórica de una serie de tiempo para hacer predicciones. Este tipo de modelo es una técnica básica de pronóstico que puede utilizarse como base para modelos más complejos. Los modelos ARIMA fueron creados por Box y Jenkins, y en sí, tienen la ventaja de que toman en cuenta la dependencia existente entre los datos, es decir, que cada observación puede ser modelada en función de valores anteriores.

La metodología de Box y Jenkins se compone de 4 fases:

- **Fase 1.** Identificación del posible modelo ARIMA que sigue la serie, esta fase demanda lo siguiente:

Decidir qué transformaciones se deben aplicar para convertir la serie observada en una serie estacionaria.

Determinar un modelo ARMA para la serie estacionaria, es decir, estimar los órdenes de  $p$  y  $q$  de su estructura autorregresiva y de media móvil.

- **Fase 2.** A partir del modelo estimado en la fase 1, se estiman los parámetros AR y MA del modelo por medio de la máxima verosimilitud y se obtienen sus errores estándar y los residuos del modelo.
- **Fase 3.** Esta fase comprende el diagnóstico, dónde se comprueba que los residuos no tienen estructura de dependencia y siguen un proceso de ruido blanco. Si los residuos muestran estructura, entonces se modifica el modelo para incorporarla y se repiten las fases anteriores hasta obtener un modelo adecuado.
- **Fase 4.** La cuarta fase comprende la predicción, es decir, utilizar el modelo para predecir valores futuros [1].

### Caso planteado

A continuación, se presenta un caso propuesto de una aplicación directa a los negocios, dónde los datos fueron obtenidos por la Universidad de Porto [2]. El caso trata sobre una empresa dedicada a la movilidad urbana, la cual ha lanzado una aplicación en internet para la renta de bicicletas. El gerente de la empresa requiere una predicción del uso de sus bicicletas para el siguiente año para definir si se comprarán más bicicletas o si mejor invertirá el dinero en una campaña publicitaria.

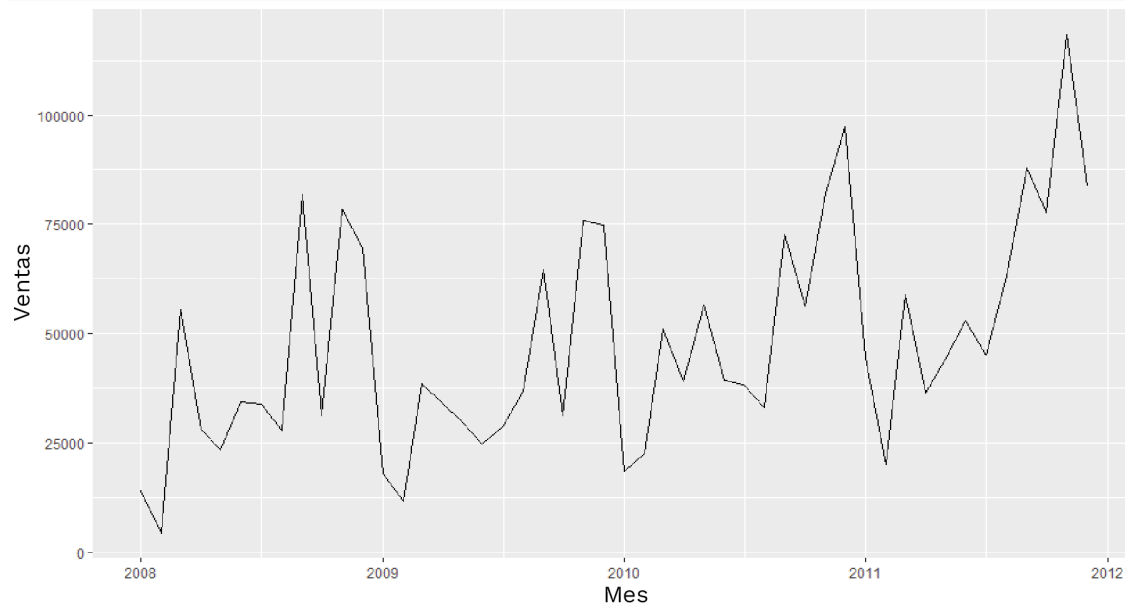


Figura 1. Uso de las bicicletas por día desde 2011 hasta finales de 2012.

Consideremos los datos de uso de las bicicletas por día de acuerdo a la serie de tiempo mostrada en la **Figura 1**. Por inspección visual, se nota que el uso diario de las bicicletas es muy variante, sin embargo, si se puede encontrar un primer patrón de comportamiento, y es que, en los meses de verano, el uso de las bicicletas incrementa, y esta demanda cae bruscamente en los meses de invierno.

Los cambios drásticos de día a día pueden afectar o sesgar el modelo, por lo cual, lo primero que se debe hacer es eliminar los valores atípicos de la serie de tiempo. En **R**, la función **tsclean** realiza dicha función, de manera que reemplaza las observaciones atípicas mediante un suavizado. Esta función también es capaz de ingresar valores perdidos en la serie si es que faltan algunos. El **Código 1** muestra la manera de aplicar la función en R, y la **Figura 2** muestra la serie limpia.

#-----

#### #Código 1. Limpieza de la serie de tiempo.

```
count_ts = ts(Info_diaria[, c('cnt')])
```

```
Info_diaria$clean_cnt = tsclean(count_ts)
```

ggplot() +

```
geom_line(data = Info_diaria, aes(x = Date, y = clean_cnt)) + labs(x = "Mes", y = "Uso
diario de bicicletas")print(Ventas.ts)
```

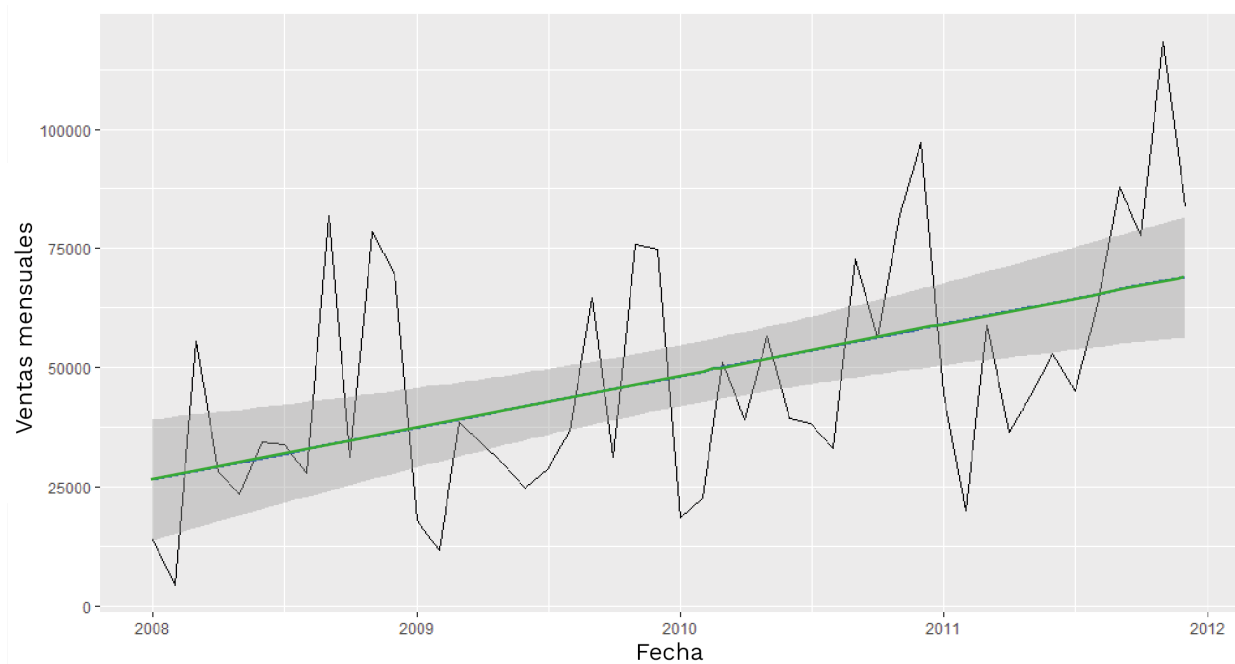


Figura 2. Serie temporal limpia, con valores atípicos eliminados.

A pesar de haber eliminado varios valores atípicos, las observaciones entre día y día cambian mucho, afectando la estimación de algún modelo a implementar. Lo más conveniente es entonces suavizar la serie de tiempo utilizando el **promedio móvil**. Esta es una forma simple de obtener la tendencia de la señal al promediar ciertas observaciones a lo largo de varios períodos de tiempo, por lo que suaviza los datos observados en una serie predecible más estable.

**Ojo!!**, la media móvil en este contexto es distinta del componente  $MA(q)$  en la definición del modelo ARIMA. El promedio móvil  $MA(q)$  como parte del marco ARIMA se refiere a retrasos de errores y combinaciones, mientras que el resumen estadístico de promedio móvil se refiere a una técnica de suavizado de datos. El **Código 2** muestra una manera de aplicar el promedio móvil en R. La **Figura 3** muestra el resultado obtenido.

#-----

**#Código 2. Aplicación del promedio móvil.**

```
Info_diaria$cnt_ma = ma(Info_diaria$clean_cnt, order=7) # using the clean count with
no outliers
```

```
Info_diaria$cnt_ma30 = ma(Info_diaria$clean_cnt, order=30)
```

```
ggplot() +
```

```
  geom_line(data = Info_diaria, aes(x = Date, y = clean_cnt, colour = "Uso diario")) +
```

```
  geom_line(data = Info_diaria, aes(x = Date, y = cnt_ma, colour = "Promedio semanal
de uso")) +
```

```
  geom_line(data = Info_diaria, aes(x = Date, y = cnt_ma30, colour = "Promedio
mensual de uso")) + ylab('Uso de bicicletas')+xlab('Fecha')
```

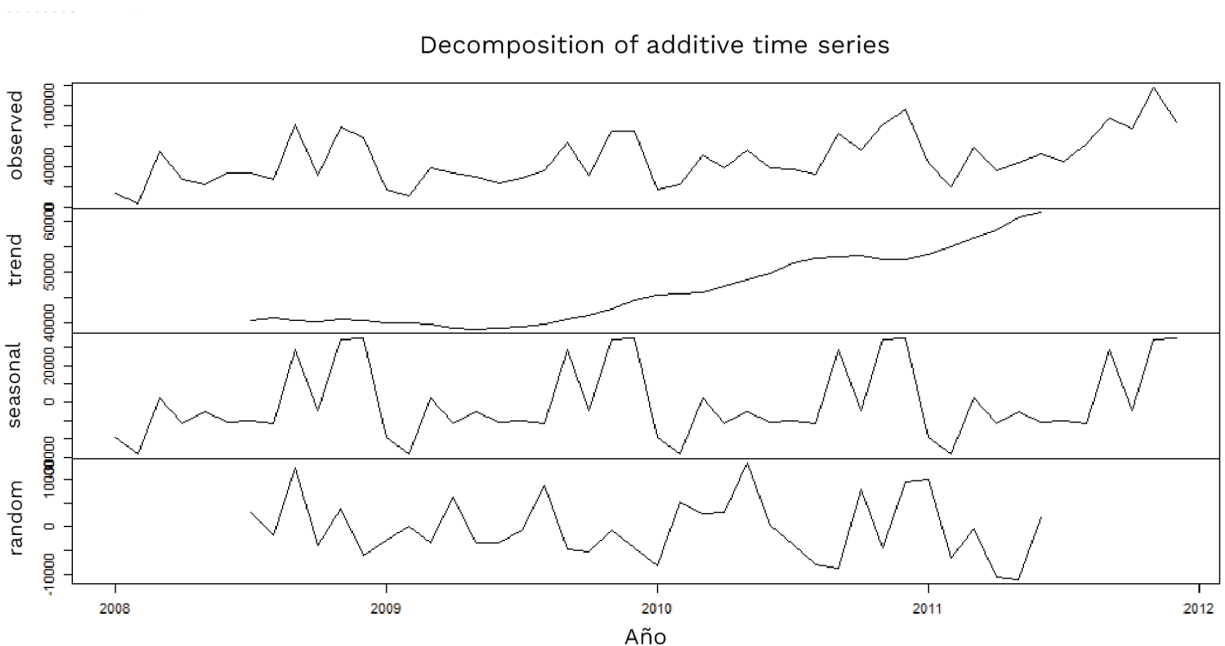


Figura 3. Tendencia de la serie temporal por semana y por mes.

Con el promedio semanal o mensual será entonces más verídica la predicción del modelo. El siguiente paso es la descomposición de la señal serie en sus tres componentes: la **tendencia**, la **estacional** y su **componente aleatoria**. La descomposición de la serie se puede realizar de 2 maneras:

Modelo aditivo

$$Y = S_1 + T_1 + E_1$$

Modelo multiplicativo

$$Y = S_1 T_1 E_1$$

Donde  $S_1$  es la componente estacional,  $T_1$  es la tendencia y ciclo de la serie,  $E$  es el error restante y  $Y$  es el número de bicicletas alquiladas. **Un modelo aditivo suele ser más apropiado cuando el componente estacional o de tendencia no es proporcional al nivel de la serie, ya que solo podemos superponer (es decir, agregar) componentes para reconstruir la serie.** Por otro lado, si el componente de estacionalidad cambia con el nivel o la tendencia de la serie, una simple "superposición" o adición de componentes, no será suficiente para reconstruir la serie. En ese caso, un modelo multiplicativo podría ser más apropiado.

Como se vio en los recursos de la semana, los modelos ARIMA se ajustan a observaciones estacionales y no estacionales. El ARIMA estacional requiere una especificación más complicada de la estructura del modelo, aunque el proceso de determinación de sus 3 parámetros es similar al de elegir los parámetros de orden no estacionales. Por lo tanto, se utiliza para este caso un modelo ARIMA no estacional.

### Eliminando la estacionalidad

El primer paso para estimar el modelo ARIMA es calcular el componente estacional de los datos usando la función **stl**. Esta función es flexible, ya que permite descomponer y pronosticar la serie. Calcula el componente estacional de la serie utilizando el suavizado y ajusta la serie original restando la estacionalidad, tal como se muestra en el **Código 3. La Figura 4** muestra las componentes de la serie, mostrando que en efecto se eliminó la estacionalidad.

#-----

#### #Código 3. Eliminando la componente estacional

```
count_ma = ts(na.omit(Info_diaria$cnt_ma), frequency=30)
decomp = stl(count_ma, s.window="periodic")
deseasonal_cnt <- seasadj(decomp)
plot(decomp)
```

**Ojo!!**, **stl** por default asume la estructura del modelo aditivo. Si fuese necesario cambiar el tipo de modelo es necesario colocar la siguiente línea **allow.multiplicative.trend=TRUE** para incorporar el modelo multiplicativo.

En el caso de la estructura del modelo aditivo, la misma tarea de descomponer la serie y eliminar la estacionalidad se puede lograr simplemente restando el componente estacional de la serie original, sin embargo, la función **seasadj** es un método conveniente si se utilizan paquetes.

En cuanto al parámetro de frecuencia en **ts**, se está especificando la periodicidad de los datos, es decir, el número de observaciones por período. Dado que estamos utilizando datos diarios suavizados, tenemos 30 observaciones por mes.

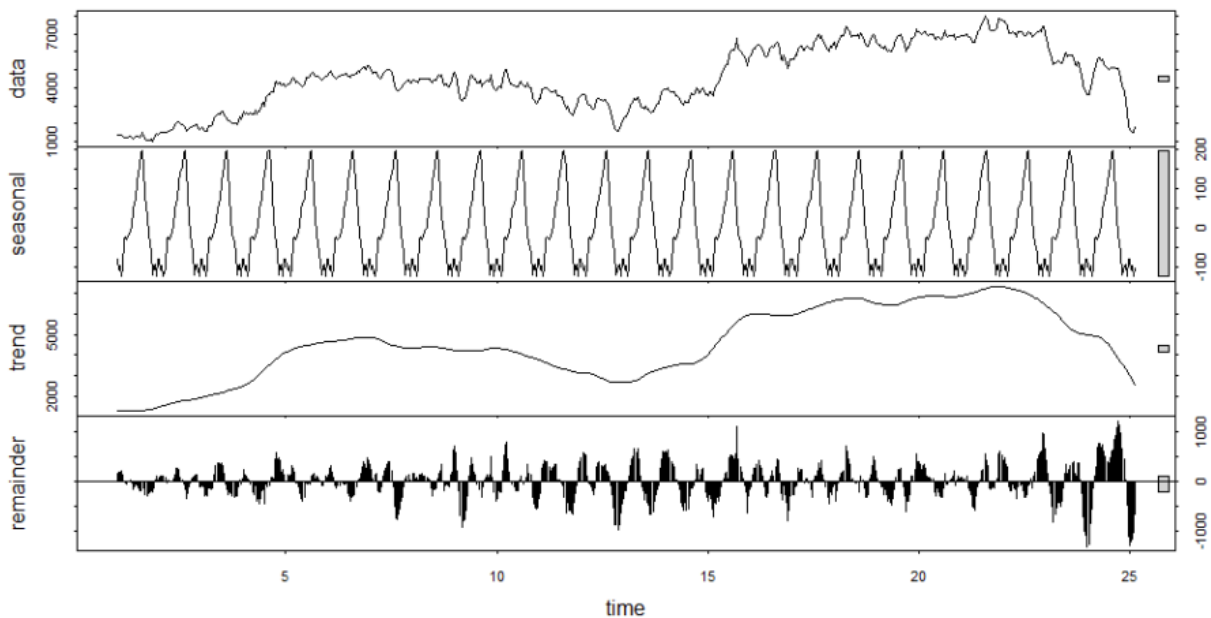


Figura 4. Descomposición de la serie temporal con la estacionalidad eliminada.

## Eliminando la tendencia

El siguiente paso es eliminar la tendencia, una forma de hacerlo es a través de la diferenciación. El **Código 4** muestra la manera de hacerlo, y la **Figura 5** muestra la serie temporal final con la tendencia eliminada.

#-----

#### #Código 4. Eliminando la componente estacional

```
count_d1 = diff(deseasonal_cnt, differences = 1)
plot(count_d1)
```

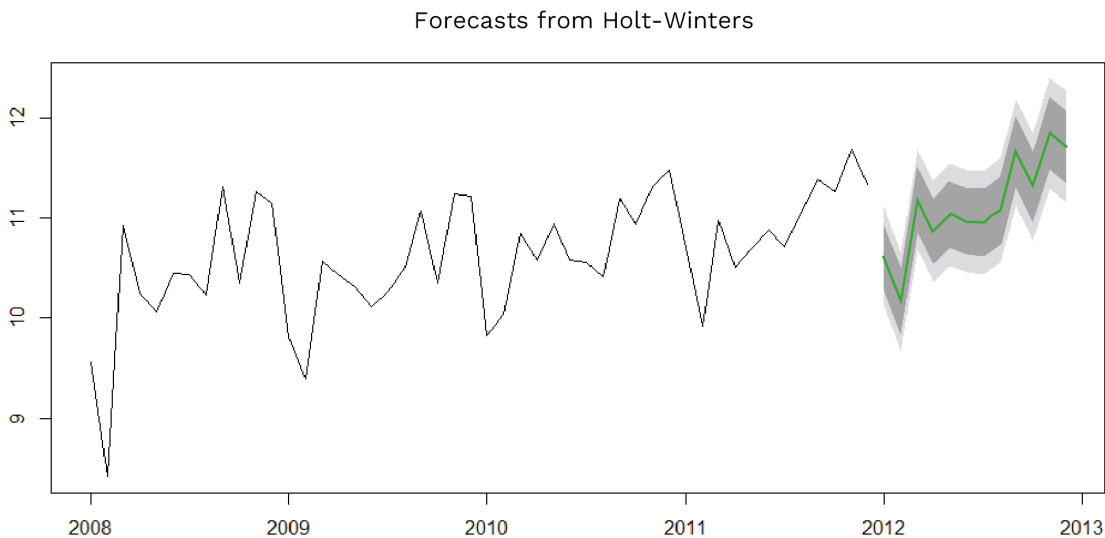


Figura 5. Serie temporal de bicicletas con tendencia eliminada.

### Ajuste del modelo ARIMA

En R, con el paquete **forecast** se puede indicar el modelo a utilizar por medio de la función **arima**, o se puede estimar automáticamente usando la función **auto.arima**. La función busca a través de combinaciones de parámetros de orden y elige el conjunto que optimiza los criterios de ajuste del modelo. El **Código 5** muestra la manera de estimar una predicción de 30 valores al futuro, la **h** es la manera en que se indica cuantos datos se desean predecir.

#-----

#### #Código 5. Estimación del modelo

```
fit_w_seasonality = auto.arima(deseasonal_cnt, seasonal=TRUE)
fit_w_seasonality
seas_fcast <- forecast(fit_w_seasonality, h=30)
```

```
plot(seas_fcast)
```

A continuación, se muestran los parámetros del modelo que se obtienen al ejecutar el Código 5, el resultado muestra 2 términos de auto regresión y 2 de media móvil.

```
Series: deseasonal_cnt
ARIMA(2,1,2)(1,0,0)[30]

Coefficients:
          ar1      ar2      ma1      ma2      sar1
s.e.  1.3644 -0.8027 -1.2903  0.9146  0.0100
s.e.  0.0372  0.0347  0.0255  0.0202  0.0388

sigma^2 estimated as 24810:  log likelihood=-4688.59
AIC=9389.17  AICC=9389.29  BIC=9416.68
```

$$\hat{Y}_t = 1.3644Y_{t-1} - 0.8027Y_{t-2} - 1.2903e_{t-1} + 0.9146e_{t-2} + E$$

Finalmente, la **Figura 6** muestra la predicción final. Las estimaciones muestran límites de confianza: 80% para los sombreados en azul oscuro y 95% en azul claro. Recuerda que los pronósticos a más largo plazo tienen más incertidumbre. El patrón en los límites de confianza puede indicar la necesidad de un modelo más estable. En predicción, siempre es muy importante observar los límites de la predicción y tener en cuenta el error esperado asociado con las estimaciones puntuales.

### Forecasts from ARIMA(2,1,2)(1,0,0)[30]

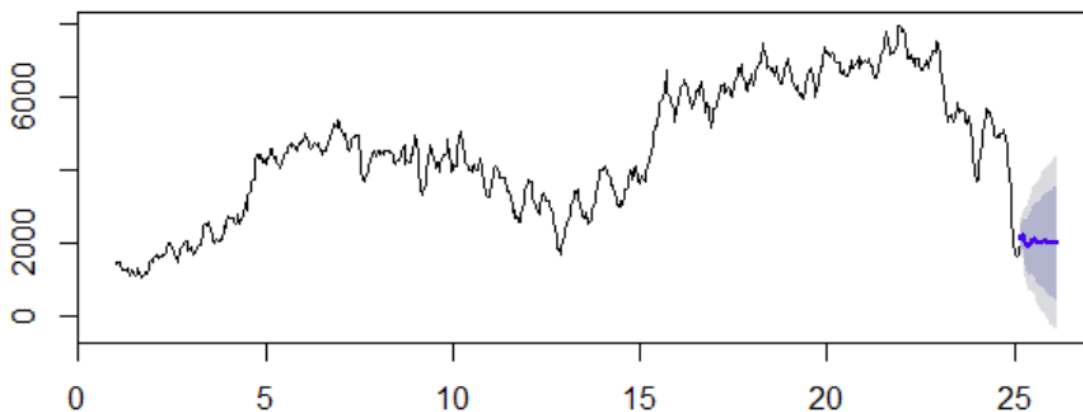


Figura 6. Predicción del uso de bicicletas.

### Referencias

- (2019). Retrieved from <http://www.estadistica.net/ECONOMETRIA/SERIES-TEMPORALES/modelo-arima.pdf>
- Fanaee-T, Hadi, and Gama, Joao, 'Event labeling combining ensemble detectors and background knowledge', Progress in Artificial Intelligence (2013): pp. 1-15, Springer Berlin Heidelberg,

